LEVEL II (12)

CENTER FOR CYBERNETIC STUDIES

The University of Texas
Austin, Texas 78712

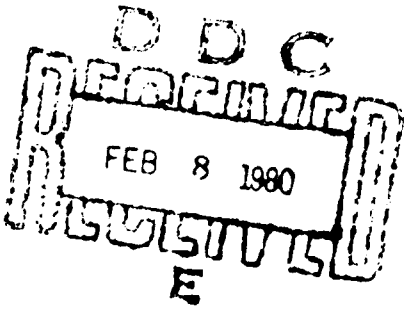
See 1473

Research Report CCS 354

A LINEAR PROGRAMMING ALGORITHM FOR CURVE FITTING IN THE L_∞ NORM

by

R. D. Armstrong*
M. G. Sklar**



October 1979

*The University of Texas at Austin
**St. Edwards University, Austin

This research was partly supported by Project NR047-021, ONR Contract N00014-75-C-0569 with the Center for Cybernetic Studies, The University of Texas at Austin. Reproduction in whole or in part is permitted for any purpose of the United States Government.

CENTER FOR CYBERNETIC STUDIES

A. Charnes, Director
Business-Economics Building, 203E
The University of Texas at Austin
Austin, TX 78712
(512) 471-1821

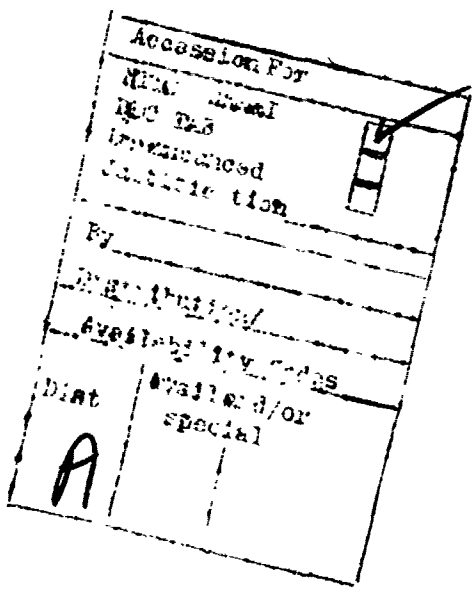
A LINEAR PROGRAMMING ALGORITHM FOR CURVE FITTING IN THE L_∞ NORM

Abstract

The L_∞ norm has been widely studied as a criterion for curve fitting problems. This paper presents an algorithm to solve discrete approximation problems in the L_∞ norm. The algorithm is a special-purpose linear programming dual method which employs a reduced basis and multiple pivots. Results of the computational experience with a computer code version of the algorithm are presented.

R. D. Armstrong, Associate Professor
University of Texas at Austin
Austin, Texas 78712

M. G. Sklar, Assistant Professor
St. Edwards University
Austin, Texas 78704



1. Introduction and Problem Statement.

The L_∞ norm problem, also called minimax and the Chebychev problem, has been widely studied as a criterion for curve fitting. This procedure minimizes the maximum residual, and is particularly well suited to problems found in numerical analysis (Rabinowitz (1968)), where a typical application arises when values are truncated to a fixed number of decimal places; and the resulting errors due to round-off are assumed to have an underlying uniform distribution. Of particular interest in numerical analysis is the error between the approximation of a function and the true function. It is desirable to place a bound on the error, and in the absence of a priori knowledge of what the numbers involved will be, it seems reasonable to allow for the worst case. Then the goal is to utilize a procedure, for approximating the function, which produces the minimum maximum error, hence the minimax or L_∞ norm method (see Ralston (1965)).

Although other procedures are more popular for "statistical data analysis," the L_∞ norm may be useful in identifying outliers (Sposito (1976)). Stiefel (1959) developed a method called the "exchange method" for finding L_∞ norm estimates, and he later (1960) established the equivalence of the method to the simplex procedure. Harter (1975) and Stiefel (1964) discuss the historical development of the L_∞ norm estimation procedure, and Appa and Smith (1973) identify a number of important properties.

It is generally accepted that some form of linear programming

algorithm provides the most efficient way to determine the L_∞ norm estimates. The purpose of this paper is to present a specialized dual linear programming algorithm for obtaining a Chebychev approximation to an overdetermined system of linear equations for the model $\underset{\sim}{C} = A^T\underset{\sim}{\beta} + \underset{\sim}{\varepsilon}$. The algorithm uses a reduced basis, multiple pivots, and a reduced ratio test.

1.1 Statement of the problem

The general L_∞ norm problem may be characterized in the following way. Let $(c_i , a_{i1} , a_{i2} ,..., a_{im})$, $i = 1, 2,...,n$ represent the values observed during n repetitions of an experiment, where c is a vector of observed values of the dependent variable and the a's are the observed values of the independent variables (the predictor variables). The objective is to determine estimates for the parameters $\underset{\sim}{\beta} = (\beta_1, \beta_2, ..., \beta_m)$ which solve the problem:

(1) Minimize the value of

$$\lambda = \text{maximum}\{|c_i - a_{i1}\beta_1 - a_{i2}\beta_2 - ... - a_{im}\beta_m|\},$$
$$i = 1,2,...,n$$

It is a well-known result that (1) may be expressed in a linear programming formulation (L. P.) as follows:

(2) Minimize λ

subject to $$c_i - \lambda \leq \sum_{j=1}^{m} a_{ij}\beta_j \leq c_i + \lambda \ , \ i = 1,2,...,n$$

where the optimal estimates of $\underset{\sim}{\beta}$ will minimize the maximum deviation (λ).

In matrix notation, the constraints of (2) are

(3) $$c - e\lambda \leq A^T\beta \leq c + e\lambda \ ,$$

where e is a vector of ones and A^T is an n by m matrix.

It is assumed that A^T has full column rank: that is, rank $(A^T) = m$. Rank deficiencies can easily be handled within the linear programming framework (see, for instance, Ben-Israel and Charnes (1968)), and will not be reviewed here. Each interval constraint may be written as two constraints which, when combined

with the objective function, results in the following problem.

(4) Minimize λ

subject to $$A^T\beta + e\lambda \geq C$$
$$A^T\beta - e\lambda \leq C \quad .$$

The linear programming dual of (4) may be written as

(5) Maximize $C^T\pi' + C^T\pi''$.

subject to $$A\,\pi' + A\,\pi'' = 0$$
$$e^T\pi' - e^T\pi'' = 1$$
$$\pi' \geq 0 \ , \ \pi'' \leq 0 \ .$$

Problem (5) may, of course, be solved using the traditional simplex method, which will be discussed in the next section. This discussion is not meant to be a complete description of linear programming, but rather is intended to establish notation and terminology. A short summary of the simplex method follows.

Given a basic feasible solution, corresponding to an extreme point of the solution space, a nonbasic variable is selected to enter the basis. (In this paper the usual terminology will be employed, where a variable is said to enter (or leave) the basis when, in fact, it is the vector associated with the variable which enters (or leaves) the basis, for the vector space R^{m+1}.) The variable selected to enter the basis is a variable whose corresponding constraint in the dual of the problem being discussed violates feasibility. (In the example of this paper, for an entering variable of problem (5), the corresponding constraint of problem (4) would be infeasible.) The variable to leave the basis is selected by determining the minimum of a set of ratios,

where the leaving variable is the first variable to become infeasible as the value of the entering variable changes.

The new set of basic variables differs by one from the preceeding set, and a new extreme point (ignoring degeneracy) in the solution space is represented by the basis. Since a finite number of extreme points exist in the solution space, and an iteration as described above moves from one extreme point to an adjacent (improving) extreme point, the algorithm will converge to an optimal solution in a finite number of steps. The convergence difficulties that arise when degeneracy is present will not be discussed in this paper. The reader is referred to Charnes (1952) for a discussion of the resolution to the problem of degeneracy. In the next section, the steps of the general algorithm described here will be discussed further.

2. General Linear Programming Approach

Prior to describing the general L. P. approach and the specific algorithm presented here, a number of terms are defined. (Note that the omission of a prime or double prime on a term which usually has one or the other indicates the term applies to both the prime and the double prime cases.)

The n constraints of (2) are denoted as "variable interval" (VI) constraints. This terminology arises from the fact that the bounds on the interval may be expanded by increasing λ.

n = the number of variable interval constraints in the primal problem (2);

$A._j$ = the m-dimensional column vector for the j-th column of A, where $A._j$ is the same for both π'_j and π''_j ;

$\hat{A}'._j$ = the (m+1)-dimensional augmented column vector for π'_j, where a +1 is appended to the vector $A._j$;

$\hat{A}''._j$ = the augmented vector for π''_j, where a -1 is appended to the vector $A._j$.

$k(i)$ = the index for the i-th basic variable, $i = 1,2,\ldots,$ $m+1$. That is, if either π'_r or π''_r is basic at the i-th position, $k(i) = r$;

F = the $(m+1)$ by $(m+1)$ basis for the dual problem (5) ;

b = the original right hand side for the dual problem (5), $b = (0, 0,\ldots,0,1)^T$;

$\hat{b}$ = the $(m+1)$-dimensional updated right hand side of (5), $\hat{b} = F^{-1}b$;

C_F^T = an $(m+1)$-dimensional vector of the original objective function coefficients for the basic variables of problem (5) ;

C_j = the original objective function coefficient in the dual problem (5) for π'_j and for π''_j

$\bar{c}'_j$ = the reduced cost for π'_j ;

$\bar{c}''_j$ = the reduced cost for π''_j ;

$y'_{\cdot j}$ = the representation of $\hat{A}'_{\cdot j}$ in terms of the basis F, so that $y'_{\cdot j} = F^{-1}\hat{A}'_{\cdot j}$;

$y''_{\cdot j}$ = the representation of $\hat{A}''_{\cdot j}$, $y''_{\cdot j} = F^{-1}\hat{A}''_{\cdot j}$;

NB' = $\{j|\pi'_j$ is nonbasic$\}$, the index set for nonbasic variables of the type π' ;

NB'' = $\{j|\pi''_j$ is nonbasic$\}$

NB = $NB' \cup NB''$

λ = the value of the objective function;

β = the parameters to estimate in the primal problem (4).

2.1 Steps of the general L. P. simplex method.

Given a basic feasible solution, the steps of the general L. P. simplex problem may be stated as follows.

Step 1. Compute the reduced costs for nonbasic variables;

a. $\bar{c}'_j = C_j - C_F^T F^{-1} \hat{A}'_{\cdot j}$, for π'_j , $j \in NB'$

b. $\bar{c}''_j = C_j - C_F^T F^{-1} \hat{A}''_{\cdot j}$, for π''_j , $j \in NB''$

c. select the variable to enter corresponding to

$$\max_{j \in NB} \{\bar{c}'_j , \bar{c}'_j > 0 \; ; \; -\bar{c}''_j , \bar{c}''_j < 0\}$$

d. if $\bar{c}'_j \leq 0$ and $\bar{c}''_j \geq 0$ for all j, terminate with the optimal solution;

Step 2. Compute the representation for π_s, the entering variable

a. $y'_{\cdot s} = F^{-1} \hat{A}'_{\cdot s}$ for π'_s entering the basis ;

b. $y''_{\cdot s} = F^{-1} \hat{A}''_{\cdot s}$ for π''_s entering the basis ;

Step 3. Compute the updated right hand side

a. $\hat{b} = F^{-1} b$

Step 4. Select as the variable to leave the basis that variable corresponding to the minimum ratio

a. if π'_s is the entering variable,

$$\min \left\{ \frac{\hat{b}_j}{y'_{js}} \right\} , \; \frac{\hat{b}_j}{y'_{js}} > 0 \qquad , j = 1, 2, \ldots, m+1$$

b. if π''_s is the entering variable,

$$\min \left\{ -\frac{\hat{b}_j}{y''_{js}} \right\} , \; \frac{\hat{b}_j}{y''_{js}} < 0 \qquad , j = 1, 2, \ldots, m+1$$

c. if the minimum occurs for $j = r$, $\pi_{k(r)}$ leaves the basis

Step 5. Update F^{-1} , $C_F^T F^{-1}$, $k(r)$, NB' and NB". Go to Step 1.

3. Reduced Basis Structure

This section discusses how the full tableau data may be represented with a reduced basis. The traditional approach of the prior section for solving the L. P. problem (5) may be altered in the following ways. Instead of using the (m+1) by (m+1) matrix F to solve the system of equations, an m by m basis may be isolated to perform the usual simplex steps. The primary purpose of this paper is to show how the reduced basis may be efficiently used to solve (5).

Throughout the paper, a distinction will be made between two problem structures: the full structure which has (m+1) basic variables, this structure corresponds to the regular simplex tableau; and the reduced structure with m basic variables, which are the first m of the basic variables in the full structure. Consequently, there are two representations for a nonbasic variable. One representation is in terms of the (m+1) basic variables of the full structure and the other is in terms of the m basic variables of the reduced structure.

Several advantages become evident when the reduced basis structure is used. The usual simplex steps may be performed in terms of the reduced basis, and a reduced minimum ratio procedure is presented which provides a simpler ratio formulation, and which may result in fewer ratios computed than in the full tableau. Multiple pivots may be performed, where a multiple pivot allows a movement to an adjacent extreme point without updating the basis.

3.1 Partitioning

The proposal is to construct a reduced basis with which to perform the simplex operations. Since the basis, F, for the full structure (5) has linearly independent columns, it may be partitioned in the following manner:

B = the m by m basis for the reduced structure. The columns of B are those columns of A corresponding to the first m basic variables of the full structure;

G = the m-dimensional column vector of A corresponding to the (m+1)-st basic variable in the full structure, so that $G = A_{\cdot k(m+1)}$

D = the row vector for the (m+1)-st row of F, so that

$$D_i = \begin{cases} 1 \text{ if } \pi'_{k(i)} \text{ is basic} \\ -1 \text{ if } \pi''_{k(i)} \text{ is basic} \end{cases} \quad \text{for } 1 \leq i \leq m;$$

f = the (m+1)-st element of the column $\hat{A}_{\cdot k(m+1)}$ associated with the (m+1)-st basic variable in the full structure, so that

$$f = \begin{cases} 1 \text{ if } \pi'_{k(m+1)} \text{ is basic} \\ -1 \text{ if } \pi''_{k(m+1)} \text{ is basic} \end{cases}$$

Note that appending D_i to $B_{\cdot i}$ gives $\hat{A}_{\cdot k(i)}$, i = 1,2,...,m, and appending f to G gives $\hat{A}_{\cdot k(m+1)}$.

Then

$$F = \begin{pmatrix} B & G \\ D & f \end{pmatrix},$$

where B is m by m, D is 1 by m, G is m by 1, and f is a scalar.

Let $F^{-1} = \begin{pmatrix} M & N \\ E & q \end{pmatrix}$, and $FF^{-1} = I$.

Then,

$$\begin{pmatrix} B & G \\ D & f \end{pmatrix} \begin{pmatrix} M & N \\ E & q \end{pmatrix} = \begin{pmatrix} \underset{\sim}{I} & \underset{\sim}{0} \\ \underset{\sim}{0} & 1 \end{pmatrix}.$$

The components of F^{-1} may be written as

$$M = (B - Gf^{-1}D)^{-1} \qquad N = -MGf^{-1} = -(B - Gf^{-1}D)^{-1}Gf^{-1}$$

$$E = -f^{-1}DM \qquad q = f^{-1} - f^{-1}DN.$$

3.2 Updated Right Hand Side

The right hand side values for the basic variables in the full structure may be described as follows:

$\hat{b}_i$ = the value for $\pi_{k(i)}$, i = 1,2,...,(m+1) ,

where $\hat{b} = F^{-1}b$.

To facilitate the partitioning, $\hat{b}$ may be separated as $\hat{b} = \begin{pmatrix} \bar{b} \\ W \end{pmatrix}$, where $\bar{b}$ is the first m entries of $\hat{b}$ and W is the (m+1)-st entry of $\hat{b}$. The goal is to write $\hat{b} = (\bar{b}\ \ W)^T$ in terms of the reduced basis B. It has been previously noted that $F^{-1}b = \hat{b}$, so $F\hat{b} = b$. Then, using the partitioning of F from the previous section, and the separation of $\hat{b}$ into $\bar{b}$ and W ,

$$\begin{pmatrix} B & G \\ D & f \end{pmatrix} \begin{pmatrix} \bar{b} \\ W \end{pmatrix} = \begin{pmatrix} \underline{0} \\ 1 \end{pmatrix} = b .$$

Performing the multiplication,

$$B\bar{b} + GW = \underline{0}$$

$$D\bar{b} + fW = 1 .$$

However, $G = A_{\cdot k(m+1)}$, so that

$$B^{-1}G = B^{-1}A_{\cdot k(m+1)} = \bar{a}_{\cdot k(m+1)}$$

where $\bar{a}_{\cdot k(m+1)}$ is defined to be the m-dimensional representation for $\pi_{k(m+1)}$ in terms of the reduced basis B. For notational convenience, let

$$R = \bar{a}_{\cdot k(m+1)}$$

To continue,

$$B\bar{b} = GW = \underline{0} ,$$

$$\bar{b} = B^{-1}GW = -B^{-1}A_{\cdot k(m+1)}W = -\bar{a}_{\cdot k(m+1)}W$$

$$\bar{b} = -RW.$$

Also,

$$D\bar{b} + fW = 1 \rightarrow D(-RW) + fW = 1 \rightarrow W(f - DR) = 1 .$$

For notational convenience, define

$$sumr = (f - DR) ,$$

so $W(f - DR) = 1$, $W(sumr) = 1$,

thus $W = 1/sumr$.

To summarize, the right hand side values for the basic variables, written in terms of the reduced basis B, are

$$\begin{pmatrix} \bar{b} \\ W \end{pmatrix} = \begin{pmatrix} -RW \\ W \end{pmatrix} = \begin{pmatrix} -R/sumr \\ 1/sumr \end{pmatrix} .$$

3.3 Selection of the Entering Variable

The reduced cost, $\bar{c}'_j$ or $\bar{c}''_j$, for a variable may be described as the change in the objective function value per unit change in the value of the entering variable. The convergence of the simplex algorithm depends only on selecting a variable to enter the basis which has a positive rate of change; however, the implementation used here selects as the entering variable that variable which has the largest rate of change.

The reduced costs may be computed by

$$\bar{c}'_j = C_j - C_F^T F^{-1} \hat{A}'_{:j}$$
$$\bar{c}''_j = C_j - C_F^T F^{-1} \hat{A}''_{:j} ,$$

and the largest rate of change corresponds to

$$\max_{j \in NB} \left\{ \bar{c}'_j , \bar{c}'_j > 0 ; -\bar{c}''_j , \bar{c}''_j < 0 \right\} .$$

The optimality conditions are $\bar{c}'_j \leq 0$ and $\bar{c}''_j \geq 0$, for every $j \in NB$, in which case the algorithm terminates with the optimal solution. The optimality conditions for this dual problem (5) are related to the feasibility conditions for the primal problem (4), in the sense that selecting the entering variable with the largest rate of change in λ is tantamount to bringing feasibility to the primal constraint which is most infeasible with the current solution.

3.4 Representation of the Entering Variable

Suppose the method of the previous section has selected π_s as the entering variable. The following introduces additional notation.

$\hat{a}_{.s}$ = the first m elements in the representation for π_s, which is $y_{.s}$, in terms of the full basis F ;

V = the (m+1)-st element in the representation for π_s in terms of the full basis F ;

$S = \bar{a}_{.s}$ = the m-dimensional representation for π_s in terms of the reduced basis B, so that $B^{-1}A_{.s} = \bar{a}_{.s}$.

Since the augmented column vectors for π'_s and π''_s , $\hat{A}'_{.s}$ and $\hat{A}''_{.s}$, differ only in the last element, which is defined to be g , then either $y'_{.s}$ or $y''_{.s}$ (the representations in terms of the full basis F for π'_s and π''_s) may be partitioned as $(\hat{a}_{.s}\ V)^T$, where the sign of g determines which of $\hat{A}'_{.s}$ or $\hat{A}''_{.s}$ is being represented.

Then using the partitioned version of F, and separating $y_{.s}$ into $(\hat{a}_{.s}\ V)^T$, we have

$$\begin{pmatrix} B & G \\ D & f \end{pmatrix} \begin{pmatrix} \hat{a}_{.s} \\ V \end{pmatrix} = \begin{pmatrix} A_{.s} \\ g \end{pmatrix} ,$$

so that the element g may be defined as

$$g = \begin{cases} 1 & \text{if } \pi'_s \text{ is entering the basis} \\ -1 & \text{if } \pi''_s \text{ is entering the basis} \end{cases}$$

Performing the multiplication,

$$B\hat{a}_{.s} + GV = A_{.s}$$

$$D\hat{a}_{.s} + fV = g .$$

Solving the first equation for $\hat{a}_{.s}$,

$$\hat{a}_{.s} = B^{-1}(A_{.s} - GV) = B^{-1}A_{.s} - B^{-1}GV$$

$$\hat{a}_{.s} = B^{-1}A_{.s} - B^{-1}A_{\cdot k(m+1)}V$$

$$\hat{a}_{.s} = \bar{a}_{.s} - \bar{a}_{k(m+1)}V \ .$$

Solving the second of the equations,

$$D\hat{a}_{.s} + fV = g$$

$$D(S - RV) + fV = g$$

$$DS + V(f - DR) = g \ .$$

Recalling that $W = 1/(f - DR) = 1/sumr$,

then $DS + V/W = g$

$$V/W = g - DS \ .$$

Define $sums = g - DS$,

then $V/W = sums$,

$$V = sums/sumr \ .$$

Therefore, the representation for π_s in terms of the full structure is $y_{.s} = (\hat{a}_{.s}\ V)^T$, which has now been written in terms of the reduced basis, so that

$$\begin{pmatrix} \hat{a}_{.s} \\ W \end{pmatrix} = \begin{pmatrix} S - RV \\ V = sums/sumr \end{pmatrix} \ .$$

4. Simplification with the reduced structure

Solving the dual problem (5) using the reduced basis B offers several advantages, which will be discussed in this section.

The original primal problem (3) had n interval constraints which were separated into 2n constraints, so the dual problem (5) has 2n structural variables. However, there are n pairs of variables, π'_j and π''_j, for which the $A_{\cdot j}$ vector is identical, and the augmented vectors of $A_{\cdot j}$, $\hat{A}'_{\cdot j}$ and $\hat{A}''_{\cdot j}$, differ only in the entry in the (m+1)-st (last) element. To exploit this pairing, the algorithm computes reduced costs for n-(m+1) rather than for 2n-(m+1) nonbasic variables. For any pair π'_j and π''_j the reduced costs differ by 2λ, since $\hat{A}'_{\cdot j}$ and $\hat{A}''_{\cdot j}$ differ only in the sign of the "1" in the last entry.

Additionally, selection of the variable to leave the basis may be simplified using the reduced structure. The minimum of ratios of the form R_i/S_i will designate the leaving variable, and some of the ratios may be ignored, since it may be anticipated they will not be the minimum.

Also, a multiple pivot may be performed, which includes a "pivot" where the basis is not changed in the reduced structure (although in the full structure a basis change would occur). This exchange of variables does not affect the basis and therefore results in a savings in computation.

The reduced structure has m basic variables but the (m+1)-st basic variable in the full structure, $\pi_{k(m+1)}$, will be treated as a pseudo-basic variable in the reduced structure. For convenience, the variable will be denoted by $\pi_{k(m+1)}$ in the reduced structure just as it is in the full structure. This variable will play a role in the reduced structure as though it were basic, in the sense that a variable, π_s, entering the basis in the reduced structure will replace a current basic variable if possible, and otherwise will replace $\pi_{k(m+1)}$. If $\pi_{k(m+1)}$ is replaced by π_s, no basis change occurs in the reduced structure, although in the full structure an explicit basis change would occur. Treating the variable $\pi_{k(m+1)}$ as a pseudo-basic variable in this fashion enables the same extreme point path to be followed in the reduced structure as is followed in the full structure.

4.1 Computing the Reduced Costs

Using the reduced structure may result in some simplification when determining the reduced costs of the dual problem (5). To see this, we consider the primal problem (4). In essence, the procedure will be to compare the i-th residual from the fitted plane (for the current values of β) to the current value of λ. This is analagous to computing, in the primal problem (4), the amount of infeasibility of a particular constraint.

To aid in the discussion of the computation of the reduced costs, the following terms are defined.

$\bar{c}_j$ is the reduced cost for the j-th nonbasic variable, which may correspond to either π'_j or π''_j, depending on the sign of the residual;

$h_j = c_j - \hat{\beta}^T A_{\cdot j}$, so that h_j is the j-th residual from the fitted line $\underset{\sim}{a}\hat{\underset{\sim}{\beta}} = (a_1\hat{\beta}_1 + a_2\hat{\beta}_2 + \ldots + a_m\hat{\beta}_m)$.

then,

a. $\bar{c}_j = |c_j - \hat{\beta}^T A_{\cdot j}| - \lambda$,

$\bar{c}_j = |\ h_j\ | - \lambda$.

If $h_j > 0$, $\bar{c}_j$ is the reduced cost for π'_j ;

If $h_j < 0$, $-\bar{c}_j$ is the reduced cost for π''_j ;

1. suppose $h_j > 0$, so only π'_j will be a candidate. This

means that the point $(a_{.j}\hat{\beta}_j, a_{1j}, a_{2j}, \ldots, a_{mj})$ on the fitted plane (which will be called the fit) is below the observed point $(c_j, a_{1j}, a_{2j}, \ldots, a_{mj})$. If $h_j - \lambda > 0$, then the fit is below the bound $c_j - \lambda$; thus, the primal constraint $c_j - \lambda \leq A^T\hat{\beta}$ is infeasible, and π'_j is the dual variable associated with this constraint. If $h_j - \lambda < 0$, the fit is within the interval bounds and the constraint is feasible.

2. For the case where $h_j = c_j - \hat{\beta}^T A_{.j} < 0$, similar reasoning suggests that $\bar{c}_j$ is the reduced cost for π''_j, and the reduced cost for π'_j may be ignored, as its associated constraint in (2) must be satisfied.

Then, the variable selected to enter the basis is the variable corresponding to

$$\max_{j \in NB' \cap NB''} \{\bar{c}_j\} \text{ , for } \bar{c}_j > 0,$$

and terminate with the optimal solution if no maximum exists. (In relating the reduced costs described above to those encountered in the full tableau, the sign of $\bar{c}_j$, when $\bar{c}_j$ refers to π''_j, will be negative in the tableau but positive in this algorithm.)

Note that when π_p is basic (either π'_p or π''_p), the reduced cost for its nonbasic complement $\hat{\pi}_p$ (either π''_p or π'_p) is not considered.

This will be discussed in detail in the multiple pivot section, where the procedure is given which determines when the complement of a basic variable should be considered a candidate to enter the basis.

4.2 Ratios Using the Reduced Basis

In this section it will be shown that utilizing the reduced basis may result in the computation of fewer ratios, and that the ratio test procedure of the reduced structure (computing ratios of the form R_i/S_i) is equivalent to the procedure of the full structure, in the sense that both procedures select the same variable to leave the basis.

Prior to the discussion of the ratios, it may be helpful to summarize some of the previous results. The following is a partial tableau for the full structure, where the entries are written in terms of the reduced basis.

Basis	π_s	RHS	Ratios
λ	$\bar{c}_s$	λ	
$\pi_{k(1)}$	S_1-R_1V	$-R_1/sumr$	$(-R_1/sumr)/(S_1-R_1(sums/sumr))$
$\pi_{k(2)}$	S_2-R_2V	$-R_2/sumr$	$(-R_2/sumr)/(S_2-R_2(sums/sumr))$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\pi_{k(m)}$	S_m-R_mV	$-R_m/sumr$	$(-R_m/sumr)/(S_m-R_m(sums/sumr))$
$\pi_{k(m+1)}$	$V=\frac{sums}{sumr}$	$1/sumr$	$1/sums$

$$sums \neq 0$$

The i-th ratio in the reduced structure is of the form R_i/S_i, and if R_t/S_t is the minimum ratio in the reduced structure, then the corresponding variable, $\pi_{k(t)}$, is selected to leave the basis.

Two variables, π_i and π_j, will be said to be the same type of variable if both are restricted to be ≥ 0 (that is, π_i' and π_j') or if both are restricted to be ≤ 0 (such as π_i'' or π_j''). The following function

is defined to determine the sign of a quantity ξ:

$$\text{sign}(\xi) = \begin{cases} 1 & \text{if } \xi > 0 \\ -1 & \text{if } \xi < 0 \end{cases}$$

The above sign function will be used to determine which of the ratios, R_i/S_i , should be candidates and which may be ignored. (For convenience, the following two statements will be considered equivalent:

(1) a ratio will be considered a candidate...; and

(2) the variable associated with a ratio will be considered a candidate...).

Recall the scalar f is the (m+1)-st element in the vector $\hat{A}_{\cdot k(m+1)}$, and f is either +1 (for $\pi'_{\cdot k(m+1)}$), or -1 (for $\pi''_{\cdot k(m+1)}$). Recall also that g is the (m+1)-st element in the vector $\hat{A}_{\cdot s}$, and g is either +1 (for $\pi'_{\cdot s}$) or -1 (for $\pi''_{\cdot s}$) . Then, the i-th ratio in the reduced problem, R_i/S_i , will be considered a candidate when

$$fg = -\text{sign}(R_i) * \text{sign}(S_i) \ , \ S_i \neq 0 \ .$$

If this condition fails, the associated ratio need not be considered (nor computed), since the ratio will not be the minimum. In the computer code, an equivalent condition must be satisfied for a ratio to be considered a candidate:

$$D_i g = \text{sign}(S_i) \ .$$

For convenience in isolating terms, the ratio

$$(-R_i/sumr)/(S_i-R_i sums/sumr)$$

will be written as

$$1/((-S_i/R_i(sumr + sums)), \quad R_i \neq 0 .$$

When $R_i = 0$, the results which follow concerning the relationship between the reduced and full ratios follow in a trivial manner.

To show that the reduced ratio test is equivalent to the regular ratio test in the full structure, four lemmas will be presented. For each lemma, only the case where $\pi'_{k(m+1)}$ is in the basis and π''_s is the entering variable will be considered. The other three cases follow similar logic. Since $\pi'_{k(m+1)}$ is basic, and variables of the type π' are restricted to be nonnegative, then $sumr > 0$, because $\pi'_{k(m+1)} = 1/sumr > 0$. Since π''_s is the entering variable, only variables associated with negative ratios will be candidates to leave the basis. Further, because $\pi'_{k(m+1)}$ and π''_s are different types, the i-th ratio in the reduced structure will be a candidate only when the signs of R_i and S_i are the same.

Lemma 1

If the i-th ratio in the reduced structure is a candidate, and the corresponding i-th ratio in the full structure is a candidate, then the (m+1)-st ratio is not the minimum ratio for the full structure.

Proof

If the (m+1)-st ratio in the full structure is not a candidate

the lemma is satisfied. If the (m+1)-st ratio is a candidate, sums is negative, since the (m+1)-st ratio is 1/sums, and only negative ratios may be considered. If the i-th ratio in the reduced structure is a candidate, the signs of R_i and S_i are the same. Then the corresponding i-th ratio of the full structure is related to the (m+1)-st ratio of the full structure by

$$\left|\frac{1}{(-S_i/R_i)\text{sumr} + \text{sums}}\right| < \left|\frac{1}{\text{sums}}\right| \quad , R_i \neq 0 \ ,$$

since both terms in the denominator of the i-th ratio are negative. This result establishes lemma 1.

Lemma 2

If the i-th ratio in the reduced structure, R_i/S_i , is not a candidate, but the corresponding i-th ratio in the full structure is a candidate, then the i-th ratio in the full structure is not the minimum ratio in the full structure.

Proof

Since the i-th reduced ratio is not a candidate, the signs of R_i and S_i are different, which means the first term in the denominator of

$$\frac{1}{(-S_i/R_i)\text{sumr} + \text{sums}}$$

is positive, and this requires sums to be negative for the i-th full ratio to be a candidate. Because sums is negative, the (m+1)-st full ratio is a candidate, so that the i-th and (m+1)-st ratios in the full problem are related by

$$\left|\frac{1}{(-S_i/R_i)\text{sumr} + \text{sums}}\right| > \left|\frac{1}{\text{sums}}\right| , \quad R_i \neq 0 \ .$$

This proves lemma 2.

Lemma 3

If the i-th ratio in the reduced structure is a candidate but the i-th ratio in the full structure is not, then the i-th ratio in the reduced structure is not the minimum of the reduced ratios.

Proof

If the i-th reduced ratio is a candidate, the signs of R_i and S_i are the same and $S_i \neq 0$, For the i-th ratio in the full structure to not be a candidate, the ratio must be positive, so that

$$\frac{1}{(-S_i/R_i)\text{sumr} + \text{sums}} > 0 \ , \ R_i \neq 0$$

The first term in the denominator is negative, which requires that sums > 0 , so that

$$(S_i/R_i)\text{sumr} < \text{sums}, \text{ and}$$

$$(R_i/S_i) > (\text{sumr}/\text{sums}).$$

Since the i-th reduced ratio is a candidate, and the i-th full ratio is not, the (m+1)-st ratio, 1/sums , cannot be a candidate because only negative ratios may be eligible, and sums is required to be positive to insure the i-th full ratio is not a candidate. Therefore, since the i-th and (m+1)-st ratios are not candidates in the full structure, some other ratio in the full structure, call it the j-th ratio, must be a candidate. Then

$$\frac{1}{(-S_j/R_j)\text{sumr} + \text{sums}} < 0, \ R_j \neq 0 \ .$$

Since sums > 0, the first term in the denominator must be negative, which requires that R_j and S_j have the same signs, which means the j-th

reduced ratio is a candidate. Because the j-th full ratio must be negative (for it to be a candidate),

$$|(-S_j/R_j)\text{sumr}| > \text{sums} ,$$

$$(S_j/R_j)\text{sumr} > \text{sums} ,$$

$$(R_j/S_j) < (\text{sums}/\text{sumr}) , R_j , S_j , \text{sumr} \neq 0 .$$

Thus the relationship between the i-th and j-th ratios of the full structure is

$$(R_j/S_j) < (\text{sums}/\text{sumr}) < (R_i/S_i)$$

and this result proves the lemma.

Lemma 4

Suppose the i-th and j-th ratios in the reduced structure are candidates, and the corresponding i-th and j-th ratios in the full structure are candidates. If the absolute value of the i-th reduced ratio is smaller than the absolute value of the j-th reduced ratio, then the same relation holds in the full structure.

Proof

Let $|R_i/S_i| < |R_j/S_j|$, which means for the case being considered that $R_i/S_i < R_j/S_j$, since the ratios are positive. For the full structure, the corresponding i-th and j-th ratios are

$$\left|\frac{1}{(-S_i/R_i)\text{sumr} + \text{sums}}\right| \quad \text{and} \quad \left|\frac{1}{(-S_j/R_j)\text{sumr} + \text{sums}}\right| .$$

Sumr is positive since $\pi'_{k(m+1)} = 1/\text{sumr}$ is basic. Since both the i-th and j-th reduced ratios are candidates, the first term in the denominator of both full ratios is negative (the signs of R_i and S_i are the same). Both full ratios are negative, since both are candidates, and it is π''_s which is entering the basis. Therefore, for each full ratio,

either sums is negative, or sums is positive, but less than the absolute value of the first term in the denominator. In either case, sums may be ignored.

Since $R_i/S_i < R_j/S_j$, then $S_i/R_i > S_j/R_j$, so that

$|(-S_i/R_i)sumr| > |(-S_j/R_j)sumr|$,

which means that

$$\frac{1}{|(-S_i/R_i)sumr|} < \frac{1}{|(-S_j/R_j)sumr|} ,$$

and therefore the i-th and j-th full ratios are related by

$$\left|\frac{1}{(-S_i/R_i)sumr + sums}\right| < \left|\frac{1}{(-S_j/R_j)sumr + sums}\right| ,$$

which proves the lemma.

Theorem

The ratio test procedure described for the reduced structure is equivalent to the ratio test procedure for the full structure, in the sense that both procedures will select the same variable to leave the basis.

Proof.

It follows from lemmas 1 and 3 that the (m+1)-st ratio of the full structure will be the minimum ratio when none of the reduced ratios are candidates. Otherwise, from lemmas 2 and 3 it can be ascertained that only ratios which are candidates in both the reduced and full structures will provide the minimums for the respective structures. Furthermore, lemma 4 guarantees that these minimums will occur at the same position in the reduced and full structures.

4.3 Multiple pivots

The general idea of a multiple pivot is that when a variable, say π_s, enters the basis in place of a current basic variable, say π_p, the increase in the objective function value, λ, may be sufficient to cause the complement of π_p (call it $\hat{\pi}_p$) to become eligible to enter the basis. The word complement will be used in the sense that the complement of π_p' is π_p'' and vice versa. (When $\hat{\pi}_p$ becomes a candidate in this dual problem, this means the corresponding constraint in the primal problem (2), which was binding at one bound becomes infeasible at the opposite bound.) Since, for this case, $\hat{\pi}_p$ would become a candidate to enter at the next iteration if π_s enters now, the algorithm brings $\hat{\pi}_p$ into the basis immediately in place of π_p, and π_s is still a candidate to enter in place of some other basic variable. This results in a computational saving, since the representations of π_p and $\hat{\pi}_p$ are identical in terms of the reduced basis B, thus no basis change occurs for the reduced structure.

For this discussion, it is assumed that π_s has been selected to enter the basis in place of π_p (where π_s may be π_s' or π_s'', and similarly for π_p), and that $\pi_p = \pi_{k(t)}$, $1 \leq t \leq m$, the t-th basic variable. The representation for π_p in the full structure is a unit vector, but the representation for its complement, $\hat{\pi}_p$, may be obtained in a manner similar to that used to find the representation for π_s, so the discussion is abbreviated.

Let $\begin{pmatrix} B & G \\ D & f \end{pmatrix} \begin{pmatrix} \hat{a}_{\cdot p} \\ L \end{pmatrix} = \begin{pmatrix} A_{\cdot p} \\ \ell \end{pmatrix}$, where $\hat{A}_{\cdot p} = \begin{pmatrix} A_{\cdot p} \\ \ell \end{pmatrix}$

The results of the multiplication and solving for $\hat{a}._p$ and L are

$$\hat{a}._p = (P - RL) ,$$

$$L = sump/sumr ,$$

$$sump = \ell - DP ,$$

$$P = \bar{a}._p ,$$ the representation for π_p (and $\hat{\pi}_p$) in terms of the reduced basis B ,

$$\ell = \begin{cases} 1 & \text{if } \hat{\pi}_p \text{ is } \pi_p' \\ -1 & \text{if } \hat{\pi}_p \text{ is } \pi_p'' \end{cases}$$

Then, a partial tableau for the full problem, written in terms of the reduced basis B, is shown below.

Basic	$\hat{\pi}_p$	π_p	π_s	RHS
λ	$\bar{c}_p = 2\lambda$	0	$\bar{c}_s$	λ
$\pi_{k(1)}$	$P_1 - R_1 L$	0	$S_1 - R_1 V$	$-R_1/sumr$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\pi_{k(t)} = \pi_p$	$P_t - R_t L$	1	$S_t - R_t V$	$-R_t/sumr$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\pi_{k(m)}$	$P_m - R_m L$	0	$S_m - R_m V$	$-R_m/sumr$
$\pi_{k(m+1)}$	$L = \frac{sump}{sumr}$	0	$V = \frac{sums}{sumr}$	$1/sumr$

(Note that $P_i = 0$, $i \neq t$, and $P_t = 1$, since $A_{.k(t)}$ is the t-th column of B and $P = B^{-1}A_{.k(t)}$.

Since π_s has been selected to enter the basis in place of π_p, the change in λ, $\hat{\delta}$, which will result when π_s enters the basis is

$$\hat{\delta} = |\bar{c}_s/(-R_t/\text{sumr})/(S_t - R_t V)| ,$$

$$\hat{\delta} = |\bar{c}_s/\text{sums}| .$$

This movement in λ , $\hat{\delta}$, may be sufficient to cause $\hat{\pi}_p$ to become a candidate to enter the basis. The minimum change in λ , δ^* , which will enable $\hat{\pi}_p$ to become a candidate may be shown to be

$$\delta^* = 2\lambda \left((-R_t/\text{sumr})/(P_t - R_t L)\right) ,$$

which, with a few sign changes and substitutions, may be rewritten

$$\delta^* = -2\lambda R_t/(2R_t + \ell\text{sumr}) .$$

Therefore,

a. if $\hat{\delta} \leq \delta^*$, this implies that when π_s enters the basis, the change in λ is not sufficient to cause $\hat{\pi}_p$ to become a candidate to enter the basis. Thus, π_s replaces π_p in the basis and the usual update operations take place.

b. if $\delta^* < \hat{\delta}$, this implies that if π_s enters the basis in place of π_p , the change in λ would be sufficient to cause $\hat{\pi}_p$ to become a candidate to enter the basis at the next iteration. Therefore, $\hat{\pi}_p$ enters immediately in place of π_p and π_s is still a candidate to replace some other variable.

This multiple pivot strategy will not be employed when the leaving variable is $\pi_{k(m+1)}$, since if the complement of $\pi_{k(m+1)}$ is brought into the basis, the algorithm finds the nonbasic variable corresponding to the next minimum ratio, and π_s then replaces this next variable, and the basis is updated. Howe\ the work

associated with finding the next smallest ratio and then performing a pivot is more than for the case where the complement of $\pi_{k(m+1)}$ does not enter the basis, and a multiple pivot is not performed.

The multiple pivot strategy may save considerable computation. As evidenced by the results of the computational experience, the occasions for multiple pivots occur between 17% and 42% of the time, depending on the problem size.

4.4 Steps for the Reduced Structure

In this section, the steps of the algorithm discussed in prior sections are summarized. Initially assume that a basic feasible solution is at hand.

a. the parameters to estimate in the primal problem (4) are the β's, which are initially computed by $\beta = C_B^T B^{-1}$.

b. the value of the objective function, λ , may be initially computed by

$$\lambda = |\bar{c}_s/\text{sums}| \quad .$$

The initial situation is that a reduced basis for (5), consisting of m variables is obtained. The full structure has m+1 basic variables and the (m+1)-st variable, $\pi_{k(m+1)}$, is a slack variable. Recall that the variable $\pi_{k(m+1)}$ is to be treated as a pseudo-basic variable in the reduced structure, and at this juncture, the variable to be called $\pi_{k(m+1)}$ in the reduced structure has not been selected. On the basis of reduced costs, suppose π_s is determined to be eligible to enter the basis. Instead of replacing a current basic variable in the reduced structure, π_s will become the variable called $\pi_{k(m+1)}$, and λ must be adjusted to make this variable "basic." In the full structure, however, the entering variable π_s does replace the basic slack (artificial) variable in the (m+1)-st position. Since initially the

first m basic variables in the full structure equal zero (and $\pi_{k(m+1)}$ equals one), either $\pi'_{k(t)}$ or $\pi''_{k(t)}$, $1 \leq t \leq m$, may be selected to be basic. By proper choice of $\pi'_{k(t)}$ or $\pi''_{k(t)}$ as basic variables, the ratio for the slack variable may be forced to be minimum, so that the entering π_s will replace the slack $\pi_{k(m+1)}$ as desired. After the slack variable leaves the basis, the iterative process begins.

Step 1. Compute the reduced costs

a. $\bar{c}_j = |c_j - \hat{\beta}^T A_{\cdot j}| - \lambda = |h_j| - \lambda$.

1. if $h_j > 0$, $\bar{c}_j$ is the reduced cost for π'_j
2. if $h_j < 0$, $-\bar{c}_j$ is the reduced cost for π''_j,

b. determine

$$\max_{j \in NB' \cap NB''} \{\bar{c}_j\} \text{ for } \bar{c}_j > 0$$

and let the variable associated with this stipulation be π_s, the variable to enter the basis. If $\bar{c}_j \leq 0$, for all j, terminate with the optimal solution.

Step 2. Compute the reduced ratios to determine the leaving variable.

Compute:

$$\min \{R_i/S_i\} \ , \ i = 1,2,\ldots,m \ ,$$

and consider R_t/S_t a candidate if $fg = -\text{sign}(R_t) * \text{sign}(S_t)$, and $S_t \neq 0$.

There are two cases which may occur:

a. if the minimum ratio is R_t/S_t , $\pi_{k(t)}$ leaves the reduced basis, $1 \leq t \leq m$;

b. if none of the ratios is a candidate, the entering variable replaces the pseudo-basic variable $\pi_{k(m+1)}$, and no basis change occurs. In the full structure, of course, an explicit basis change occurs as π_s replaces the basic variable $\pi_{k(m+1)}$.

Step 3. Update the value of $\bar{\lambda}$.

Let $\bar{\lambda}$ be the current value of λ and π_s has been selected as the entering variable. There are three cases to consider.

a. If the variable leaving the basis (of either the reduced or full structure) is $\pi_{k(t)}$, $1 \leq t \leq m$, then

$$\bar{\lambda} \leftarrow \bar{\lambda} + |\bar{c}_s(-R_t)/(S_t\text{sumr} - R_t\text{sums})|$$

b. If the variable being replaced is $\pi_{k(m+1)}$,

$$\bar{\lambda} \leftarrow \bar{\lambda} + |\bar{c}_s/\text{sums}| \ .$$

c. For the case where a multiple pivot occurs, the process for updating λ is more complicated. Suppose the complement of a current basic variable enters the basis, say $\hat{\pi}_p$ replaces π_p (which in the full structure would be

described by $k(t) = p$, $1 \leq t \leq m$.

1. When the exchange occurs ($\hat{\pi}_p$ replacing π_p), λ is updated by computing

$$\bar{\lambda} \leftarrow \bar{\lambda} - 2\bar{\lambda}R_t/(2R_t + \ell sumr),$$

and c_s is updated by

$$\bar{c}_s \leftarrow \bar{c}_s - \delta^*|sums - (S_t/R_t)sumr|.$$

2. After this exchange and updating, the iteration is still not complete since the variable π_s, which was to have replaced π_p, must now enter the basis in place of some other basic variable. To effect this, the next smallest ratio is determined (π_p had the minimum ratio previously), and the variable associated with this next smallest ratio is replaced by π_s. Then λ is further updated to reflect this change. There are three cases to consider.

   a) If $\pi_{k(t)}$, $1 \leq t \leq m$, is replaced by π_s,
   $$\bar{\lambda} \leftarrow \bar{\lambda} + |\bar{c}_s(-R_t)/(S_t sumr - R_t sums)| ;$$

   b) If π_s replaces the pseudo-basic variable $\pi_{k(m+1)}$ in the reduced structure (and in the full structure π_s replaces the basic variable $\pi_{k(m+1)}$ in an explicit basis change), then
   $$\bar{\lambda} \leftarrow \bar{\lambda} + |\bar{c}_s/sums| ;$$

c) If a multiple pivot occurs here (within a multiple pivot), the process as described above repeats.

Step 3. Update the values of β , by computing

$$\hat{\beta} = (C_B^T \pm \bar{\lambda})B^{-1} .$$

Go to Step 1.

5.0 Computational Results

The special-purpose algorithm for obtaining L_∞ norm estimates of the parameters for the model $\underset{\sim}{C} = A^T\underset{\sim}{\beta} + \underset{\sim}{\varepsilon}$ is compared to the published algorithm of Barrodale and Phillips (1975). Our algorithm has been coded in FORTRAN as a callable subroutine, which requires an n-dimensional array and an n by m matrix for the input data, an m by m matrix for the LU decomposition, and six m-dimensional arrays.

An LU decomposition procedure is used to solve the system of equations, and since much of the original data are preserved, the incidence of round-off error is diminished. All runs were performed on a CDC 6600 with a sixty bit word, the tolerance value for zero was set at 1.E-8, and the runs were made within a few minutes of each other, so the machine load was approximately the same. The reported times were from using the MNF compiler, although the algorithms were tested using the FTN and RUN compilers as well. The IMSL Library was utilized to generate an array (n-dimensional) for the dependent variable C, and an m by n matrix A, containing data which were randomly drawn from a uniform distribution. For each problem size (each combination of m and n), the reported time is an average of 5 problems. The results are summarized in table 1. Our code is denoted AS and the Barrodale and Phillips code is denoted BP.

TABLE 1: A summary of computational testing comparing the special-purpose dual method algorithm (AS) and the Barrodale and Phillips algorithm (BP) is given. Five problems were solved at each level and the reported times are the means of the results in milliseconds, using a CDC 6600. The average number of iterations is also reported.

Problem Size		Execution Time (milliseconds)		Ratio	Average Number of Iterations	
m	n	AS	BP	BP/AS	AS	BP
5	200	88	408	4.64	8.6	14.6
	400	183	830	4.54	9.8	15.0
	600	317	1479	4.67	11.8	18.0
	800	378	1835	4.85	10.6	16.6
	1000	530	2441	4.61	11.4	17.8
10	200	429	1090	2.54	22.0	25.6
	400	714	2687	3.76	26.4	31.8
	600	957	4528	4.73	26.0	36.0
	800	1285	5721	4.45	27.2	34.0
	1000	1769	7749	4.38	28.2	36.8
15	200	1163	2820	2.42	34.0	49.6
	400	1820	6345	3.49	38.2	56.0
	600	2469	9859	3.99	40.6	58.2
	800	3421	13,460	3.93	46.0	59.4
	1000	4049	17,867	4.41	47.6	63.4
20	200	2733	4303	1.57	49.0	60.0
	400	3863	9936	2.57	53.0	69.4
	600	5542	16,585	2.99	61.8	77.4
	800	6755	23,128	3.42	64.4	81.0
	1000	8206	29,312	3.57	67.0	82.2

Conclusions:

In this paper we have presented an algorithm and FORTRAN code for determining L_∞ norm parameter estimates for the curve fitting model $\underset{\sim}{C} = A^T\underset{\sim}{\beta} + \underset{\sim}{\varepsilon}$. This algorithm is a special-purpose linear programming dual method, and the code is compared to the special-purpose code of Barrodale and Phillips.

We have indicated how to partition the A matrix to form a compact or reduced basis, and how the usual simplex tableau entries may be written in terms of this reduced basis. Special ratios are developed for determining the vector to leave the basis, which are of a simplier structure than the usual simplex ratios. A multiple pivot is developed, where movement is made to a new extreme point solution without requiring the usual basis update computations.

Appa, G. and Smith, C. (1973). On L_1 and Chebyshev Estimation, Mathematical Programming 5, pp. 73-87.

Barrodale, I., and Phillips, C.(1975). Solution of an Overdetermined System of Linear Equations in the Chebychev Norm, ACM Transactions on Mathematical Software, Vol 1, No. 3, pp.264-270.

Ben-Israel, and Charnes, A. (1968). An Explicit Solution of a Special Class of Linear Programming Problems, Operations Research 16, pp. 1166-1175.

Charnes, A. (1952). Optimality and Degeneracy in Linear Programming. Econometrica 20, p160-170.

Harter, H. L. (1975b). The Method of Least Squares and Some Alternatives IV. Int. Statist. Rev. 43, pp. 125-190 and 273-278.

Rabinowitz, P. (1968). Application of Linear Programming to Numerical Analysis, SIAM REV., 10, pp. 121-159.

Ralston, Anthony (1965). A First Course in Numerical Analysis, McGraw Hill, New York, N. Y.

Sposito, V. A. (1976). Minimizing the Maximum Absolute Deviation, SIGMAP Newsletter, No. 20, pp. 51-53.

Stiefel, E. (1959). Uber diskrete und lineare Tschebyscheff - Approximation. Numerische Mathematic , pp. 1-28.

Stiefel, E. (1960). Note on Jordan Elimination, Linear Programming and Tschebyscheff Approximation, Numerische Mathemetik, No. 2, pp. 1-17.

Stiefel, E. (1964). Methods - Old and New - for Solving the Tchebycheff Approximation Problem, SIAM Journal on Numerical Analysis I, pp. 164-176.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Center for Cybernetic Studies
The University of Texas at Austin

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP
Infinity

3. REPORT TITLE
(6) A Linear Programming Algorithm for Curve Fitting in the L_∞ Norm

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*
(10) R.D. Armstrong [redacted] M.G. Sklar

(12) 46

6. REPORT DATE
(11) Oct [redacted] 79

7a. TOTAL NO. OF PAGES
43

7b. NO. OF REFS
11

8a. CONTRACT OR GRANT NO.
(13) N00014-75-C-0569

b. PROJECT NO.
NR047-021

9a. ORIGINATOR'S REPORT NUMBER(S)
(14) CCS-354

c.

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

d.

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES
(9) Research Reports

12. SPONSORING MILITARY ACTIVITY
Office of Naval Research (Code 434)
Washington, DC

13. ABSTRACT

The L_∞ norm has been widely studied as a criterion for curve fitting problems. This paper presents an algorithm to solve discrete approximation problems in the L_∞ norm. The algorithm is a special-purpose linear programming dual method which employs a reduced basis and multiple pivots. Results of the computational experience with a computer code version of the algorithm are presented.

sub-infinity

DD FORM 1 NOV 65 1473 (PAGE 1)
S/N 0101-807-6811

Unclassified 406 197
Security Classification

A-31408

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Chebyshev Curve Fitting						
Linear Programming						
Regression Analysis						